

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

**For: FORM PROCESSING DEVICE,  
AND FORM PROCESSING  
METHOD AND PROGRAM**

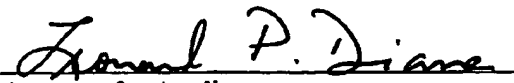
November 5, 2003

Technology Center 2100

**Applicant's undersigned attorney may be reached in our New York office by**

telephone at (212) 218-2100. All correspondence should continue to be directed to our address given below.

Respectfully submitted,

  
Attorney for Applicant

Registration No. 29,296

FITZPATRICK, CELLA, HARPER & SCINTO  
30 Rockefeller Plaza  
New York, New York 10112-3801  
Facsimile: (212) 218-2200

NYMAIN378579

10/6/14 17:00 PM 03125 US  
CA

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日      2 0 0 2 年   7 月   9 日  
Date of Application:

出 願 番 号      特 願 2 0 0 2 - 2 0 0 5 2 3  
Application Number:  
[ST. 10/C]:      [ J P 2 0 0 2 - 2 0 0 5 2 3 ]

出 願 人      キヤノン株式会社  
Applicant(s):

2 0 0 3 年   7 月 2 9 日

特許庁長官  
Commissioner,  
Japan Patent Office

今 井 康 夫

出証番号   出証特 2 0 0 3 - 3 0 6 0 0 1 0

【書類名】 特許願

【整理番号】 4516109

【提出日】 平成14年 7月 9日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 9/00

【発明の名称】 帳票処理装置および帳票処理方法ならびにプログラム

【請求項の数】 9

【発明者】

【住所又は居所】 東京都大田区下丸子3丁目30番2号 キヤノン株式会社  
社内

【氏名】 高島 敬一

【特許出願人】

【識別番号】 000001007

【氏名又は名称】 キヤノン株式会社

【代理人】

【識別番号】 100076428

【弁理士】

【氏名又は名称】 大塚 康德

【電話番号】 03-5276-3241

【選任した代理人】

【識別番号】 100112508

【弁理士】

【氏名又は名称】 高柳 司郎

【電話番号】 03-5276-3241

【選任した代理人】

【識別番号】 100115071

【弁理士】

【氏名又は名称】 大塚 康弘

【電話番号】 03-5276-3241

## 【選任した代理人】

【識別番号】 100116894

## 【弁理士】

【氏名又は名称】 木村 秀二

【電話番号】 03-5276-3241

## 【手数料の表示】

【予納台帳番号】 003458

【納付金額】 21,000円

## 【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0102485

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 帳票処理装置および帳票処理方法ならびにプログラム

【特許請求の範囲】

【請求項1】 帳票内に定義されたフィールドに流し込むデータを格納しているフィールドデータファイルを読み込み、該フィールドデータファイルのデータを該帳票内のフィールドに流し込む帳票処理装置であって、

前記フィールドデータファイルのデータが、外部ファイルを指定する記述である場合、当該記述がURL形式で記述されているかを判断する判断手段と、

前記判断手段によって、URL形式で記述されていると判断された場合には、インターネットのプロトコルに従って、ネットワークサーバ内の前記外部ファイルをダウンロードし、ダウンロードした該外部ファイルのデータを、前記帳票内のフィールドに流し込む流込手段と

を備えることを特徴とする帳票処理装置。

【請求項2】 帳票内に定義されたフィールドに流し込むデータを格納しているフィールドデータファイルを読み込み、該フィールドデータファイルのデータを該帳票内のフィールドに流し込む帳票処理方法であって、

前記フィールドデータファイルのデータが、外部ファイルを指定する記述である場合、当該記述がURL形式で記述されているかを判断する判断工程と、

前記判断工程によって、URL形式で記述されていると判断された場合には、インターネットのプロトコルに従って、ネットワークサーバ内の前記外部ファイルをダウンロードし、ダウンロードした該外部ファイルのデータを、前記帳票内のフィールドに流し込む流込工程と

を備えることを特徴とする帳票処理方法。

【請求項3】 前記ネットワークサーバ内の所定のURLよりダウンロードされた外部ファイルを、該URLと対応づけてローカルドライブに保存する保存工程を更に備え、

前記流込工程は、前記判断工程により前記記述がURL形式で記述されていると判断され、かつ該URLが前記保存工程により保存された外部ファイルのURLのいずれとも一致しない場合に、前記フィールドデータファイルにおいて指定

された外部ファイル前記ネットワークサーバよりダウンロードし、ダウンロードした該外部ファイルのデータを、前記帳票内のフィールドに流し込むことを特徴とする請求項2に記載の帳票処理方法。

【請求項4】 前記流込工程は、

前記判断工程により前記記述がURL形式で記述されていると判断され、かつ該URLが前記保存工程により保存された外部ファイルのURLと一致した場合であっても、前記保存工程により保存された外部ファイルのタイムスタンプが、前記フィールドデータファイルにおいて指定された前記ネットワークサーバ内の外部ファイルのタイムスタンプより古い場合には、該ネットワークサーバ内の外部ファイルをダウンロードし、ダウンロードした該外部ファイルのデータを、前記帳票内のフィールドに流し込むことを特徴とする請求項3に記載の帳票処理方法。

【請求項5】 前記流込工程は、

前記判断工程により前記記述がURL形式で記述されていると判断され、かつ該URLが前記保存工程により保存された外部ファイルのURLと一致した場合であって、かつ前記保存工程により保存された外部ファイルのタイムスタンプが前記フィールドデータファイルにおいて指定された前記ネットワークサーバ内の外部ファイルのタイムスタンプより古くない場合には、該保存工程により保存された外部ファイルのデータを、前記帳票内のフィールドに流し込むことを特徴とする請求項3に記載の帳票処理方法。

【請求項6】 前記保存工程は、前記ダウンロードされた外部ファイルと同じ外部ファイルが既にローカルドライブに保存されている場合には、該既に保存されている外部ファイルを削除し、前記ダウンロードされた外部ファイルを前記ローカルドライブに保存することを特徴とする請求項4に記載の帳票処理方法。

【請求項7】 前記保存工程は、前記ダウンロードされた外部ファイルが所定のファイル容量より小さい場合に、前記ローカルドライブに保存することを特徴とする請求項3に記載の帳票処理方法。

【請求項8】 前記保存工程は、前記ダウンロードされた外部ファイルのファイル容量が、前記ローカルドライブの空き容量より大きい場合には、該ローカ

ルドライブに保存されている外部ファイルを削除することを特徴とする請求項3に記載の帳票処理方法。

【請求項9】 請求項2乃至8のいずれか1つに記載の帳票処理方法をコンピュータによって実現させるための制御プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、フィールドデータファイルに記述された各データ項目を、帳票内にあらかじめ定義したフィールド（フィールドデータファイルのデータを受け取るプログラム中のメモリ領域）に流し込むためのプログラムである帳票処理プログラムに関するものである。

【0002】

【従来の技術】

従来、フィールドデータファイルに記述された各データ項目を、帳票内にあらかじめ定義したフィールドに流し込むためのプログラムである帳票処理プログラムにおいて、ビットマップファイルなどのファイル型データを扱う場合、フィールドデータファイルに当該ビットマップファイルの格納先を示すパス名を記述することによって、外部ファイル（ローカルドライブまたはネットワークサーバ内に格納されたファイルのうち、フィールドデータファイル中のデータとしてその格納先のパス名が記述されており、当該パス名によって指定されるファイル）の指定を行っていた。例えば、フィールドデータファイルに、IMAGE1=c:¥data¥image.bmpといった記述を行っていた。

【0003】

【発明が解決しようとする課題】

しかし、従来の方法では、フィールドデータファイルおよびその中で記述されているファイル型データの外部ファイルが、帳票処理プログラムが動作するコンピュータのローカルドライブやアクセス可能なネットワークドライブ（これらをまとめてローカルドライブと称す、以下同じ）に存在している必要があり、昨今多くなってきているWebサーバなどのインターネット/イントラネットサーバ



上に存在するファイルを扱うことができなかった。

#### 【0004】

本発明は、上述の点に鑑みてなされたもので、インターネット／イントラネットサーバ上に存在する外部ファイルのデータを帳票内の所定のフィールドに流し込むことが可能な、帳票処理方法、帳票処理装置、および帳票処理を実行させるための制御プログラムを提供することを目的とする。

#### 【0005】

##### 【課題を解決するための手段】

上記の目的を達成するために本発明に係る帳票処理方法は以下のような構成を備える。即ち、

帳票内に定義されたフィールドに流し込むデータを格納しているフィールドデータファイルを読み込み、該フィールドデータファイルのデータを該帳票内のフィールドに流し込む帳票処理方法であって、

前記フィールドデータファイルのデータが、外部ファイルを指定する記述である場合、当該記述がURL形式で記述されているかを判断する判断工程と、

前記判断工程によって、URL形式で記述されていると判断された場合には、インターネットのプロトコルに従って、ネットワークサーバ内の外部ファイルをダウンロードし、ダウンロードした該外部ファイルのデータを、前記帳票内のフィールドに流し込む流込工程とを備える。

#### 【0006】

##### 【発明の実施の形態】

##### 〔第1の実施形態〕

以下、本発明の実施形態を図面を参照して説明する。なお、以下の説明において用いる用語をあらかじめ定義すると、「帳票」とは、枠線、固定的な図形、文字等を有し、定義されたフィールドにデータを流し込むことが可能なファイルというものとする。また、「フィールドデータファイル」とは、帳票内に定義されたフィールドに流し込むデータをデータ項目毎にわけて保存されたファイルというものとする。また、「フィールド」とは帳票にデータを流し込む際に使用する基本的な単位であり、「データ項目」とはフィールドデータファイル内のデータ

の集合を構成する要素をいうものとする。また、「ファイル型データ」とは、ファイルとして格納されたデータ（例えば、ビットマップファイルとして格納された画像データ）をいうものとする。さらに、「帳票処理プログラム」とは、帳票内に定義されたフィールドに、フィールドデータファイルよりデータを流し込むためのプログラムを指すものとする。

#### 【0007】

図1は、本発明の第1の実施形態に係る帳票処理装置のシステム構成を示すブロック図、図2乃至図6は、第1の実施形態にかかる帳票処理方法の動作の流れを示すフローチャートである。

#### 【0008】

図1において、1は中央処理装置、2はRAMなどの記憶装置、3はCRTなどの表示装置、4はキーボード、5はマウス、6は外部補助記憶装置、7は印刷装置、8はネットワーク接続装置をそれぞれ示す。外部補助記憶装置6は、プログラム等が記憶媒体からコンピュータに配給されることを示す。プログラム等が格納される記憶媒体としては、ROM、フロッピー（登録商標）ディスク、CD-ROM、ハードディスク、メモリカード、光磁気ディスクなどを用いることが可能である。図7は、記憶媒体中の帳票処理プログラムの構成を示す。

#### 【0009】

図8は、帳票処理プログラム803、フォーム情報804およびフィールド定義情報805、キャッシュ設定806、キャッシュテーブル807が記憶装置2に格納され、プログラムが実行可能となった状態のメモリマップを示す。フォーム情報は、帳票の枠線など帳票の固定的な図形や文字のレイアウト情報である。

#### 【0010】

フィールド定義情報は、フィールドデータファイルのデータを流し込む各フィールドの位置・サイズや文字属性、流し込まれるデータの型などの情報である。キャッシュ設定は、インターネット／イントラネットサーバ（これらをまとめてネットワークサーバと称す）から、インターネットのプロトコルに従ってダウンロードしたファイルをクライアントのローカルドライブのディスクにキャッシュするかどうかを表わすキャッシュフラグ、キャッシュフォルダ名、制限容量など

が設定されている。キャッシュテーブルは、キャッシュファイル名と元のURLアドレスとタイムスタンプの組み合わせのリストである。

#### 【0011】

図10は、キャッシュテーブルの例である。キャッシュファイルは、キャッシュ設定のキャッシュフォルダ名に設定してあるフォルダに保存され、キャッシュファイル名は、キャッシュフォルダ内で一意であれば何でも良い。ここでは、ランダムに生成する16進8桁の数字の文字列に、元のファイルの拡張子をつけた文字列をファイル名とする。

#### 【0012】

タイムスタンプは、キャッシュファイルを作成した時点での元ファイルの日付で、元のURLアドレスにこれより新しいファイルが存在すれば新規にキャッシュファイルを作成する必要がある。

#### 【0013】

本実施形態は、クライアント上で動作する帳票処理プログラムが、Webサーバ上のフィールドデータファイルおよびその中で記述されているファイル型データの外部ファイルをローカルドライブにダウンロードしてフォームにオーバーレイ処理する（流し込む）一例であり、図2に示すような手順で処理を行う。

#### 【0014】

まず、Webサーバ、ftpサーバなどに格納されたフィールドデータファイルを記憶装置2のワークメモリ808上にダウンロードする（ステップS201）。このフィールドデータファイルは、クライアントで動作する帳票処理プログラムでのユーザ操作、もしくはシステムのイベントなどをトリガーにして、データベースから各種データを引き出して、Webサーバやftpサーバ上に生成されるものである。通常、Webサーバやftpサーバなどからのファイルのダウンロードは非同期データ転送で行われる。

#### 【0015】

図9は、フィールドデータファイルの記述例（（A））ならびに帳票処理プログラムによる流し込み処理が行われた後の帳票（（B））を示す図である。「PICTURE=http://www.server.com/taro.jp

g」の行は、「ラベル名=ファイル名」の書式で指定されたJ P E GファイルをP I C T U R Eというラベル名を持った帳票内のフィールドに流し込んでいる。帳票処理プログラムの場合、W e bサーバで作成されたフィールドデータファイルのデータはクライアントで何度も使われるわけではなく、その処理が完了したら破棄すべきだったり、セキュリティの問題のためローカルドライブにキャッシュファイルとして残すべきでなかったりするのが普通であるため、この例ではフィールドデータファイルをキャッシュファイルとしてローカルドライブに保存していない。

#### 【0016】

フィールドデータファイルのダウンロードが成功したか判定し（ステップS 2 0 2）、失敗であればエラー処理を行って（ステップS 2 0 3）、全体の処理を終了する。成功であれば、そのサイズが0かどうか判定する（ステップS 2 0 4）。サイズが0であれば、全体の処理を終了する。サイズが0より大きければ、フィールドデータファイルの中からデータを1項目読み込む（ステップS 2 0 5）。前述の図9のフィールドデータファイルでは1行の「ラベル名=」以降から行末までが1項目のデータであるが、カンマ区切りのC S Vファイルであれば、行頭もしくはカンマから行末またはカンマまでのデータが1項目である。

#### 【0017】

次に、読み込んだ1項目のデータがファイル型データなのか判定する（ステップS 2 0 6）。この判定は、あらかじめ記憶装置2に格納してあるフィールド定義情報をもとに行う。前述の図9のフィールドデータファイルの第1行の場合、データを流し込む帳票内のフィールドは「NAME」というラベル名なので、記憶装置2に格納してあるフィールド定義情報から「NAME」というラベル名のフィールドを検索し、そのデータ型を判定する。

#### 【0018】

データ型がファイル型データであれば、そのフィールドデータファイルの指し示す外部ファイルの読み込みを行う（ステップS 2 0 7）。後述のように、この処理によって、記憶装置2のワークメモリ808上に外部ファイルの内容が格納され、そのメモリハンドルが記憶装置2のワークメモリ808上の変数f i l e

\_\_dataに格納される。

【0019】

処理中にエラーが発生していれば、変数file\_\_dataにはNULLハンドルが格納される。そして、変数file\_\_dataの指し示すメモリハンドルに格納されたデータを帳票内のフィールドに流し込む処理を行う（ステップS208）。

【0020】

ファイル型データでなければ、従来どおりのデータ型に合ったデータ流し込み処理を行う（ステップS209）。次に、フィールドデータファイル中に次のデータ項目があるか判定し（ステップS210）、次のデータ項目があれば、ステップS205へ戻り、次のデータ項目に対して処理を続ける。データ項目の流し込みが全て完了したならば、処理を終了する。

【0021】

ファイル型データの外部ファイルの読み込み処理（ステップS207）は、図3に示すような手順で処理を行う。フィールドデータファイルに記述されているパス名がインターネット/イントラネットサーバ上かを判定する（ステップS301）。パス名として、http://やftp://などで始まるURLであれば、インターネット/イントラネットサーバ上と判定する。インターネット/イントラネットサーバ上でなければ、従来の外部ファイル読み込み処理を行う（ステップS302）。

【0022】

ファイル読み込み処理では、記憶装置2のワークメモリ上に外部ファイルの内容を格納し、そのメモリハンドルを記憶装置2のワークメモリ上の変数file\_\_dataに格納する。処理中にエラーが発生していれば、変数file\_\_dataはNULLハンドルを格納する。ステップS301でパス名がインターネット/イントラネットサーバ上であれば、そのアドレスから外部ファイルをダウンロードすべきか否かの設定を行う（ステップS303）。

【0023】

この処理の結果、ダウンロードすべきと判断された場合、記憶装置2のワーク

メモリ上の変数 `flag_download` に1、ダウンロードすべきでないと判断された場合、0が格納されるので、その値を判定し（ステップS304）、1ならばパス名のURLアドレスから外部ファイルをダウンロードする（ステップS305）。

#### 【0024】

ダウンロード処理では、記憶装置2のワークメモリ上に外部ファイルの内容を格納し、そのメモリハンドルを記憶装置2のワークメモリ上の変数 `file_data` に格納する。処理中にエラーが発生していれば、変数 `file_data` にはNULLハンドルを格納する。次に、外部ファイルのダウンロードが成功したかを判定し（ステップS306）、成功ならば、ダウンロードした外部ファイルのキャッシュファイルへの保存処理を行う（ステップS307）。ダウンロードが失敗ならば、そのまま処理を終了する。また、ステップS304で `flag_download` が0ならば、キャッシュファイルの読み込み処理を行う（ステップS308）。

#### 【0025】

ダウンロードフラグ設定処理（ステップS303）は、図4に示すような手順で処理を行う。まず、記憶装置2のキャッシュ設定のフラグがONになっているか判定し（ステップS401）、ONならば、現在処理しているファイル型データのURLアドレスが記憶装置2のキャッシュテーブルに登録されていないか検索し（ステップS402）、見つかったかを判定し（ステップS403）、既に存在しているならば、URLアドレスの外部ファイルのタイムスタンプの情報を読み込み、キャッシュファイルのタイムスタンプと比較し、キャッシュファイルのタイムスタンプがURLアドレスの外部ファイルのタイムスタンプより古いかを判定し（ステップS404）、古くなければ記憶装置2のワークメモリ上の変数DLに0をセットする（ステップS405）。それ以外の場合は、DLに1をセットする（ステップS406）。変数DLはURLアドレスのファイルをダウンロードすべきかを表わす。

#### 【0026】

キャッシュファイル保存処理（ステップS307）は、図5に示すような手順

で処理を行う。まず、記憶装置2のキャッシュ設定のフラグがONになっているか判定し(ステップS501)、ONならば、現在処理しているファイル型データのURLアドレスが記憶装置2のキャッシュテーブルに登録されていないか検索し(ステップS502)、見つかったかを判定する(ステップS503)。

#### 【0027】

キャッシュテーブルから見つければ、そのURLアドレスの外部ファイルがキャッシュされているということである。見つければ、そのエントリのキャッシュファイル名の外部ファイルをキャッシュフォルダから削除し(ステップS504)、キャッシュテーブルからそのエントリも削除する(ステップS505)。

#### 【0028】

次に、記憶装置2のキャッシュ設定の制限容量が記憶装置2のワークメモリ上の変数file\_dataに格納されたメモリハンドルの内容よりも大きいか判定し(ステップS506)、小さければキャッシュできないので、処理を終了する。大きければ、キャッシュフォルダの残り容量(=制限容量-現在のフォルダサイズ)が変数file\_dataに格納されたメモリハンドルの内容よりも大きいか判定する(ステップS507)。

#### 【0029】

キャッシュフォルダの残り容量(空き容量)の方が小さければ、記憶装置2のキャッシュテーブルの中で最も古いタイムスタンプを持つエントリのキャッシュファイルを削除し(ステップS508)、キャッシュテーブルからそのエントリも削除し(ステップS509)、ステップS507に戻り、変数file\_dataに格納されたメモリハンドルの内容をキャッシュファイルへ保存できるようになるまで古いキャッシュファイルを削除する。

#### 【0030】

ステップS507で、キャッシュフォルダの残り容量の方が変数file\_dataに格納されたメモリハンドルの内容よりも大きければ、既存の他のキャッシュファイルで使われていないキャッシュファイル名を生成し(ステップS510)、そのファイル名で変数file\_dataに格納されたメモリハンドルの内容を書き込んだキャッシュファイルをキャッシュフォルダに作成し(ステップ

S511)、キャッシュファイル作成が成功したか判定し(ステップS512)、成功であれば、作成したキャッシュファイル名と元URLアドレスとタイムスタンプをセットした新しいエントリをキャッシュテーブルに作成する(ステップS513)。

#### 【0031】

キャッシュファイル読み込み処理(ステップS308)は、図6に示すような手順で処理を行う。まず、現在処理しているファイル型データのURLアドレスに対応するキャッシュファイル名を記憶装置2のキャッシュテーブルから取得する(ステップS601)。この処理に入るのは、ステップS303の処理において、現在処理しているファイル型データのURLアドレスが記憶装置2のキャッシュテーブルから見つかった場合のみであるため、ステップS601においてキャッシュファイル名は必ず取得できる。次に、取得したキャッシュファイル名の外部ファイルを読み込む(ステップS602)。キャッシュファイルの読み込み処理では、記憶装置2のワークメモリ上に外部ファイルの内容を格納し、そのメモリハンドルを記憶装置2のワークメモリ上の変数file\_\_dataに格納する。処理中にエラーが発生していれば、変数file\_\_dataにはNULLハンドルを格納する。

#### 【0032】

##### 〔第2の実施形態〕

上記実施形態では、フィールドデータファイルがインターネット/イントラネットサーバ上にある場合であったが、ローカルディスクやファイルサーバ上にある場合でも、フィールドデータファイル中にURLアドレスとして記述されたインターネット/イントラネット上のファイルを扱うことができる。

#### 【0033】

##### 〔他の実施形態〕

なお、本発明は、複数の機器(例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど)から構成されるシステムに適用しても、一つの機器からなる装置(例えば、複写機、ファクシミリ装置など)に適用してもよい。

#### 【0034】



また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

#### 【0035】

この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

#### 【0036】

プログラムコードを供給するための記憶媒体としては、例えば、フロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

#### 【0037】

また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

#### 【0038】

さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

#### 【0039】

#### 【発明の効果】

以上説明したように本発明によれば、インターネット／イントラネットサーバ上に存在する外部ファイルのデータを帳票内の所定のフィールドに流し込むことが可能となる。

**【図面の簡単な説明】**

**【図1】**

本発明の実施形態にかかる帳票処理装置のハードウェア構成を示すブロック図である。

**【図2】**

本発明の実施形態にかかる帳票処理方法によるデータ処理の流れを示すフローチャートである。

**【図3】**

本発明の実施形態にかかる帳票処理方法によるデータ処理の流れを示すフローチャートである。

**【図4】**

本発明の実施形態にかかる帳票処理方法によるデータ処理の流れを示すフローチャートである。

**【図5】**

本発明の実施形態にかかる帳票処理方法によるデータ処理の流れを示すフローチャートである。

**【図6】**

本発明の実施形態にかかる帳票処理方法によるデータ処理の流れを示すフローチャートである。

**【図7】**

本発明の実施形態にかかる帳票処理プログラムの構成を示す図である。

**【図8】**

本発明の実施形態にかかる帳票処理プログラムが実行可能となった状態のメモリマップである。

**【図9】**

本発明の実施形態にかかる帳票処理プログラムが帳票処理するフィールドデー

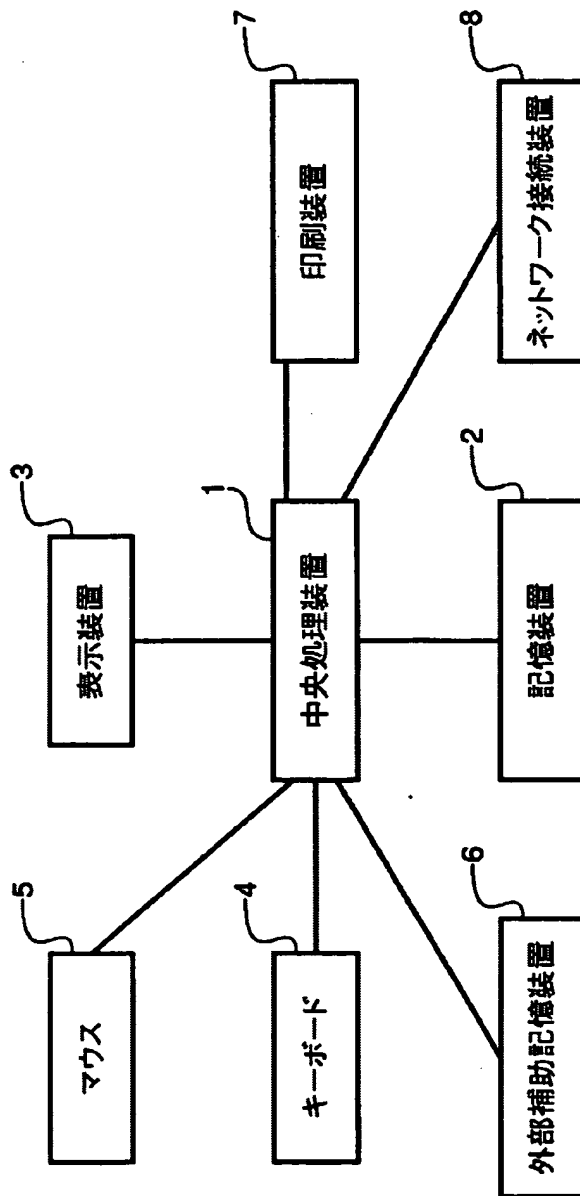
タファイルならびに帳票処理後の帳票の一例を示す図である。

【図10】

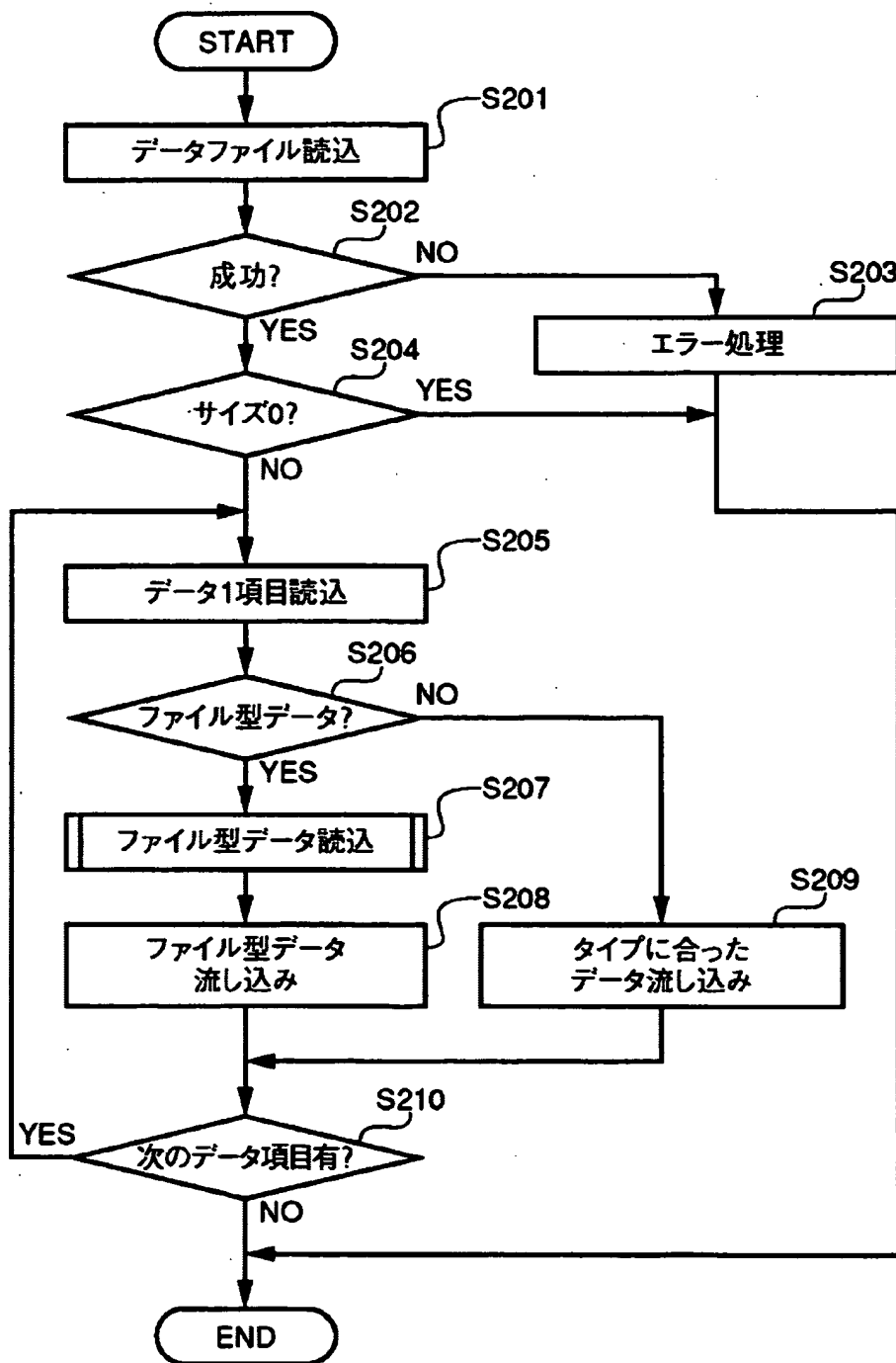
本発明の実施形態にかかる帳票処理プログラムにより生成されたキャッシュテーブルの一例を示す図である。

【書類名】 図面

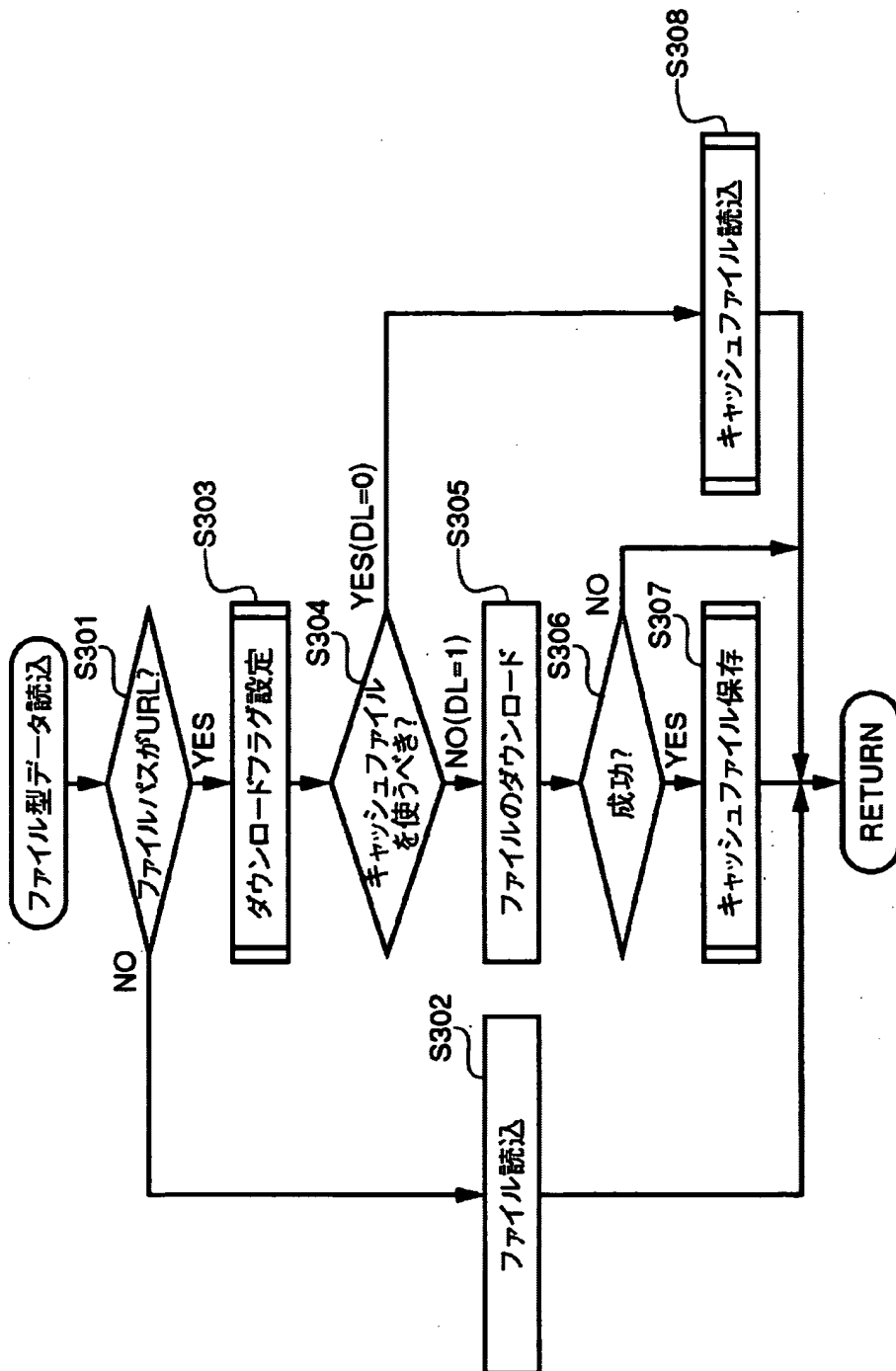
【図 1】



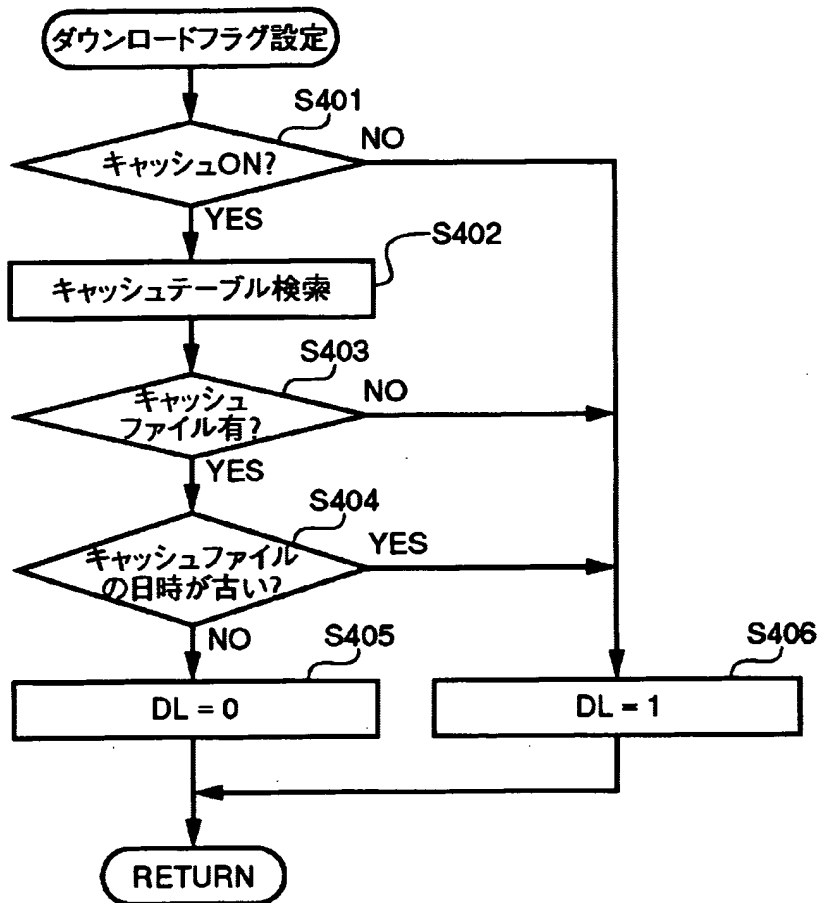
【図 2】



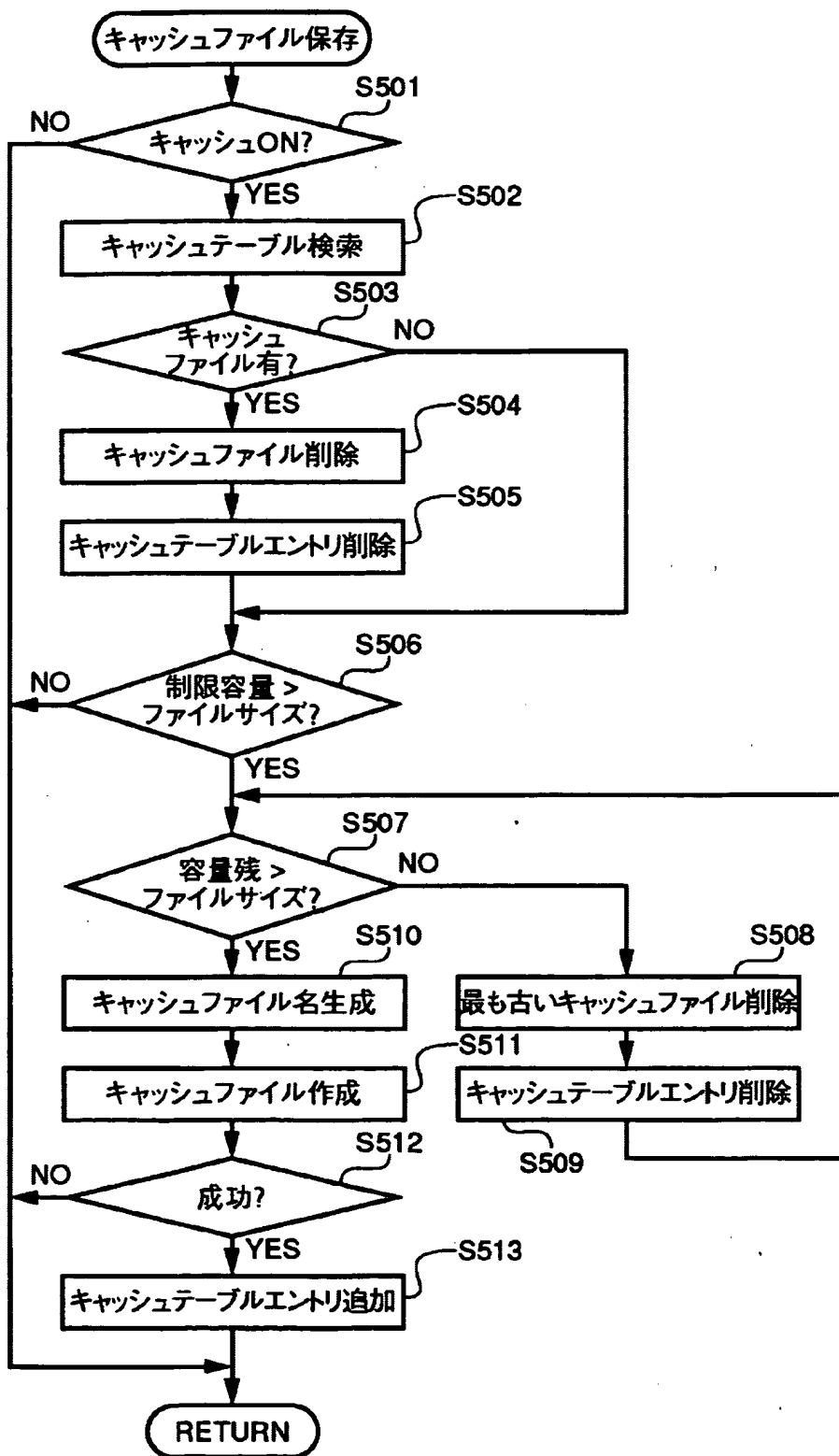
【図3】



【図4】

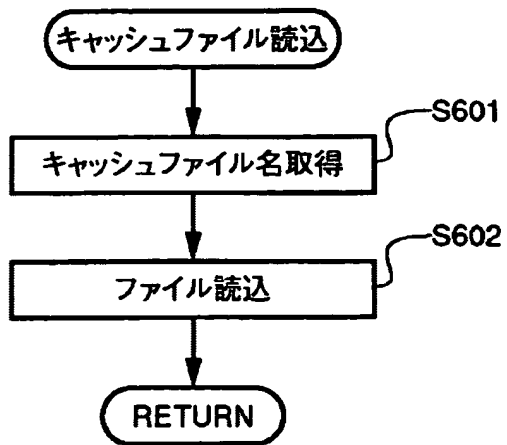


【図5】

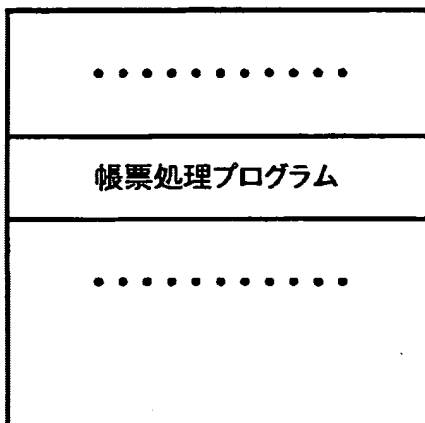




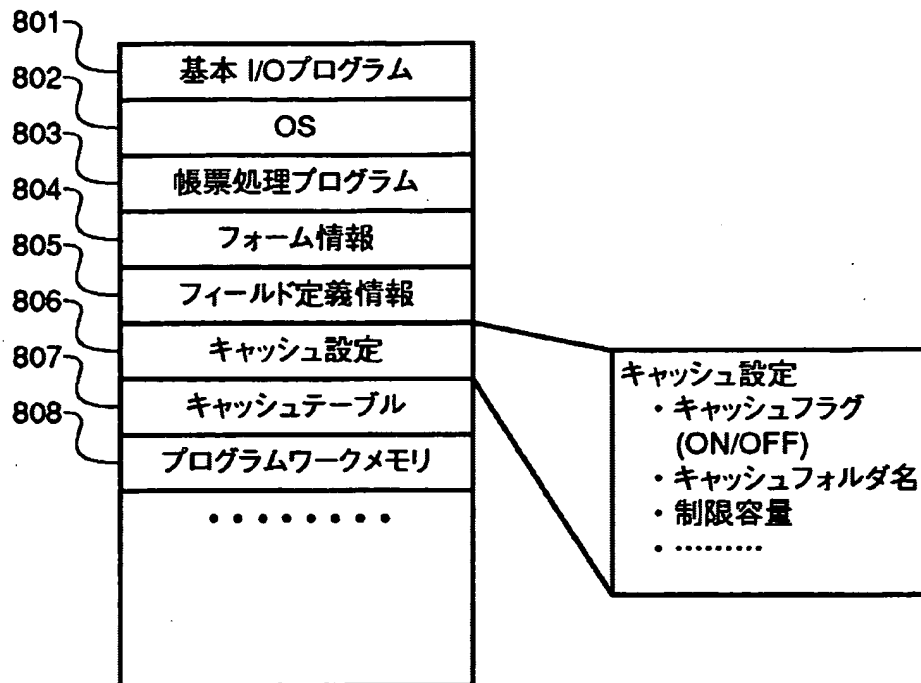
【図 6】



【図 7】



【図 8】



【図 9】

フィールドデータファイル

NAME = 帳票太郎 帳票次郎 帳票三郎  
ADDRESS = 神奈川県川崎市 神奈川県横浜市 神奈川県横浜市  
PICTURE = <http://www.server.com/taro.jpg> <http://www.server.com/jiro.jpg> <http://www.server.com/saburo.jpg>

(A)

フィールド(NAME)


フィールド(ADDRESS)

帳票処理プログラムによる  
流し込み  
↓

名前:	帳票太郎
住所:	神奈川県川崎市
	

フィールド(PICTURE)

名前:	帳票次郎
住所:	神奈川県川崎市
	

名前:	帳票三郎
住所:	神奈川県横浜市
	

帳票

(B)

【図10】

キャッシュ名	元ファイルアドレス	タイムスタンプ
00000001.jpg	http://www.server1.com/image1.jpg	2001/05/14
00000001.gif	http://www.server2.com/picture.gif	2001/06/13
00000001.bmp	http://www.server5.co.jp/table.bmp	2001/06/29
.....	.....	.....

【書類名】 要約書

【要約】

【課題】 インターネットサーバ上に存在するファイルのデータを帳票内の所定のフィールドに流し込むことを可能にする。

【解決手段】 帳票内に定義されたフィールドに流し込むデータを格納しているフィールドデータファイルを読み込み、該フィールドデータファイルのデータを該帳票内のフィールドに流し込む帳票処理方法であって、前記フィールドデータファイルのデータが、外部ファイルを指定する記述である場合、当該記述がURL形式で記述されているかを判断する判断工程（ステップS301）と、判断工程（ステップS301）によって、URL形式で記述されていると判断された場合には、インターネットのプロトコルに従って、ネットワークサーバ内の外部ファイルをダウンロードし（ステップS305）、ダウンロードした該外部ファイルのデータを、前記帳票内のフィールドに流し込む流込工程とを備える。

【選択図】 図3

特願2002-200523

出願人履歴情報

識別番号

[000001007]

1. 変更年月日  
[変更理由]

1990年 8月30日

新規登録

住 所  
氏 名

東京都大田区下丸子3丁目30番2号  
キヤノン株式会社